Cristian Herrera Ruztort et al.: Remote Control and Monitoring of VLBI Experiments by Smartphones, IVS 2012 General Meeting Proceedings, p.286–290
http://ivscc.gsfc.nasa.gov/publications/gm2012/Herrera.pdf

Remote Control and Monitoring of VLBI Experiments by Smartphones

Cristian Herrera Ruztort, Hayo Hase, Octavio Zapata, Felipe Pedreros

Transportable Integrated Geodetic Observatory - TIGO
Contact author: Cristian Herrera Ruztort, e-mail: cristian.herrera@tigo.cl

Abstract

For the remote control and monitoring of VLBI operations, we developed a software optimized for smartphones. This is a new tool based on a client-server architecture with a Web interface optimized for smartphone screens and cellphone networks. The server uses variables of the Field System and its station specific parameters stored in the shared memory. The client running on the smartphone by a Web interface analyzes and visualizes the current status of the radio telescope, receiver, schedule, and recorder. In addition, it allows commands to be sent remotely to the Field System computer and displays the log entries. The user has full access to the entire operation process, which is important in emergency cases. The software also integrates a webcam interface.

1. Introduction

In the future the new VLBI2010 antennas should be distributed more uniformly over the surface of the earth and should provide an increased observation density [1]. VLBI observations are usually monitored and controlled locally at the station. With remotely attended observations we can achieve a better telescope utilization, a faster response time on errors and maintenance procedures, and increased efficiency of observations. Nowadays smartphones, tablets, and mobile applications are very common and widely used, but for VLBI operations this is an unexplored area. We developed a new concept and tool for monitoring and controlling VLBI telescopes by smartphones called "Jmonan". This is an extension to the current NASA Field System (FS) with a modern graphical user interface optimized for smartphone screens and particulary useful for stations with a lack of operators or stations in remote locations such as TIGO or O'Higgins.

2. Different Methods of Smartphone Programming

"Mobile application development" summarizes any kind of software development for low-power handheld devices such as tablets or mobile phones. In the world of smartphone programming two very distinct branches are distinguished: native development and Web development. Native development is an application designed to run on a specific operating system and machine firmware. In many cases this type of software needs to be adapted for optimal performance on different devices, e.g., a native application developed for the iPhone will need to run on its proprietary operating system (iOS Platform). That would mean this software is not transportable to devices with other operating systems. For the development of native applications it is possible to use different programming languages such as JAVA [3] or C/C++ [4]. To become independent of the different operating systems we realized Web application development. All or some parts of the software are downloaded from a webserver each time when the software is started. The biggest

advantage is that Web applications will run on any smartphone with a modern HTML/Javascript compatible Web browser. Another feature is to introduce new versions by just upgrading the source code on the webserver and automatically providing all clients with the new version. It is not necessary to install a new application on each device. We can use languages like HTML5 [5] or Javascript for the development of this kind of application.

3. Jmonan

3.1. General Scheme

Jmonan is an acronym for **J**query [2] **mon**itoring **an**tenna . This new tool is based on Ethernet with a client-server model. Two applications communicate with each other to exchange some information: (1) the client part initiates the communication by sending service requests to the server and provides an interface to display the returning results from the server, and (2) the server part waits for requests from clients and then responds to them.

Thus the application is divided in two parts, Jmonan-Server and Jmonan-Client. The first one is hosted on the Field System PC which is behind a firewall and on a private network for security issues. The Jmonan-Server takes parameters of the Field System and of the station specific part stored in the shared memory. The actual values of these parameters are requested continuously from a webserver which is a Jmonan-Client to the Field System server and smartphone at the same time. The smartphone Web browser is served by the Web server Jmonan-client (see Figure 1).

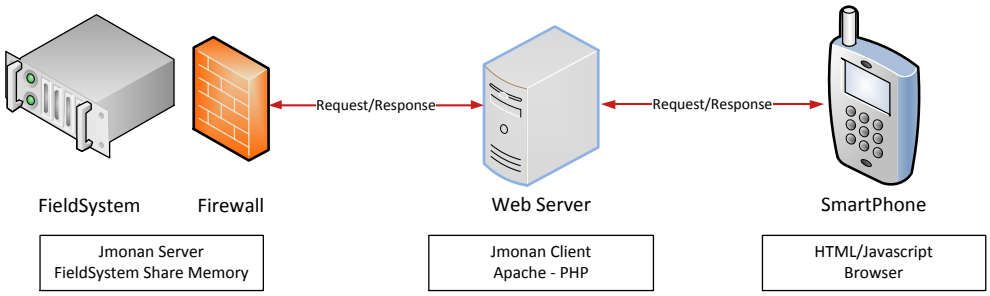

Figure 1. General scheme of Jmonan application for smartphones.

3.2. Jmonan-Server

The first part of this tool, named Jmonan-Server, is hosted on the Field System PC. This server is coded in C language with a standard protocol stack Transmission Control Protocol over Internet Protocol (TCP/IP). A concurrent and autonomous server was implemented. When the client initiates a connection to the server, it creates a new process, task, or thread. This newly created server handles the entire request of the client. On completion, this new server is terminated. In essence, each client has its own server. The advantage of such a concurrent server is that multiple clients are serviced at the same time [6]. To ensure service availability at all times, the server is checked by an additional watchdog process. Due to different tasks, the Jmonan-Server is composed of three different services:

- Monan service (Monitoring Antenna): Requests data from FS variables and station specific

variables stored in the shared memory. When the connection with the client is established, it sends each second an array with all the values in an ASCII format.

- Oprin service (Operator Input): This service accepts and executes FS commands in real time, which are sent by the client. The commands are injected into the FS using the supported injection method.
- Log service (Log Entries): Besides the parameters of the antenna and experiment, the server has an option to read and send the client the last part of the log file every second. The client process receives this data and shows the log output of the experiment in real time.

Each of these services offers different options for monitoring and controlling VLBI experiments.

3.3. Jmonan-Client

The second part of this tool, named Jmonan-Client, is part of the Web server. It was written in HTML/Javascript and PHP [7] for the communication with the server. Most Web applications must reload the entire page in order to make a request to the server. This implementation instead uses AJAX [8] (acronym for Asynchronous Javascript and XML), a technique which can send and retrieve data from a server asynchronously in the background. It avoids interference with the display and full page reloads and is therefore most efficient in terms of data transmission.

The objective of this develoment is to create a complete and flexible tool for the different tasks of controlling and monitoring VLBI experiments. The entire software was designed in a modular way, having a scalable system (see Figure 2).

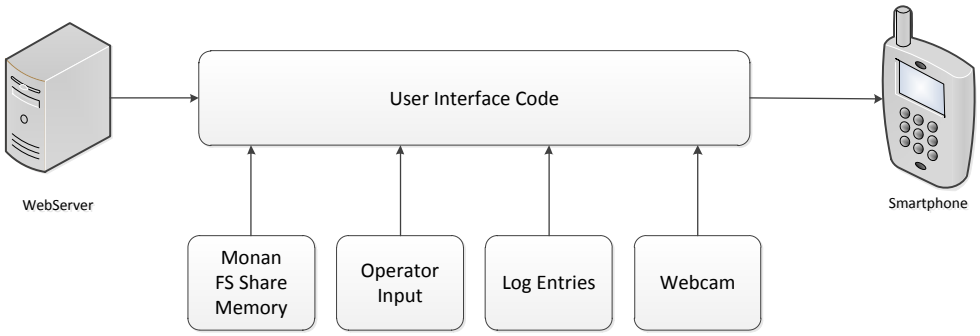

Figure 2. Jmonan-Client, modular design.

Therefore, this architecture of a Web and multiplatform interface was chosen and optimized for smartphone screens and cellphone networks. To reduce the programming effort the tool Jquery [2] (a framework of Javascript and AJAX) was used. Jquery helps to develop the user-interface for smartphone touch screens.

The following features including the display and visualization of the actual status are implemented:

- Radio telescope (mode, position, velocity, brakes). See Figure 3.
- Schedule (session, source). See Figure 3.
- Receiver (cryo parameters, box temperature, voltages).

- Recorder (recording on/off, scan name, capacity). See Figure 4.
- Webcam image.

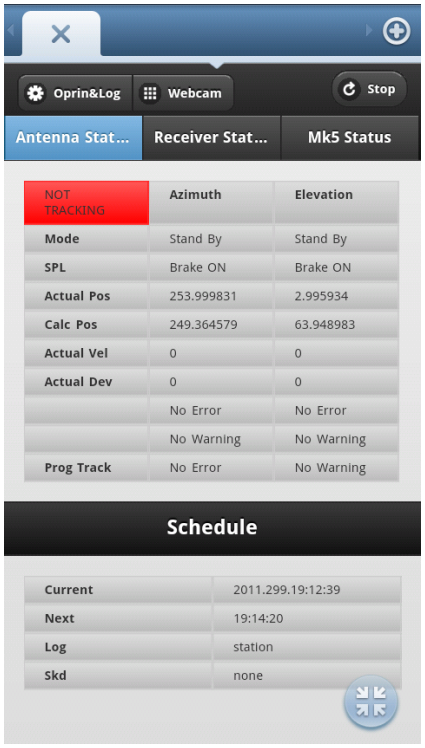

Figure 3. Jmonan-Client, antenna status monitoring.

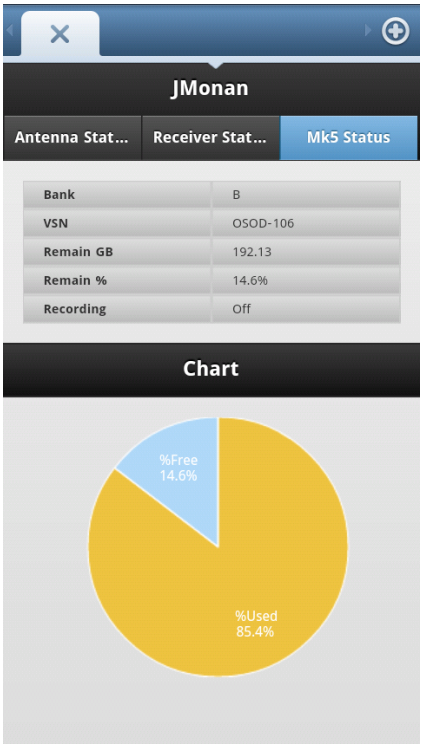

Figure 4. Jmonan-Client, Mark 5 recorder status monitoring.

To implement the remote control feature, the user can enter commands via smartphone into the FS and watch the response in the log-file, which even automatically highlights errors and warnings (see Figure 5).

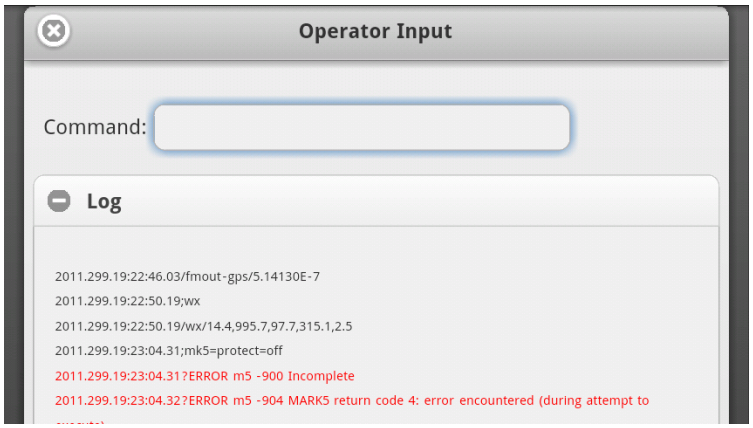

Figure 5. Jmonan-Client, operator input and log entries.

4. Security

To protect the system and proper functioning of a VLBI experiment, some security features were developed; one of them involves the HTTP authentication (login and password) of the user for the use of this application. This decreases the possibility of an unauthorized access. A second level of security of the entire system is the installation of the Jmonan-server behind a firewall and on a private network.

5. Conclusions

The Jmonan software allows VLBI operation of a specific station via smartphone. This is an advantage, especially if stations are understaffed and operators want to control the VLBI-status from home. Jmonan was used successfully during the CONT11 sessions and is used frequently at the geodetic observatory TIGO. This software also helped during the replacement procedure of an encoder, where the actual encoder reading was needed to maintain the orientation.

References

[1] Rothacher, M., Beutler, G., Bosch, W., Donnellan, A., Gross, R., Hinderer, J., Ma, C., Pearlman, M., Plag, H.-P., Richter, B., Ries, J., Schuh, H., Seitz, F., Shum, C.K., Smith, D.; Thomas, M., Velacognia, E., Wahr, J., Willis, P., Woodworth, P., The future Global Geodetic Observing System (GGOS), In: The Global Geodetic Observing System. Meeting the Requirements of a Global Society on a Changing Planet in 2020, Springer-Verlag, Plag, H.-P., Pearlman, M. (eds.), 2009.

[2] Javascript framework Jquery. http://www.jquery.com.

[3] JAVA language. http://www.java.com.

[4] Jo Stichbury, Effective C++ Programming for Smartphones (2005).

[5] W3C - World Wide Web Consortium. http://www.w3.org.

[6] W. Richard Stevens, TCP/IP Illustrated, Volume 1, The Protocols.

[7] Hypertext Preprocessor. http://www.php.net.

[8] B. Brinzarea - C. Darie, AJAX and PHP: Building Modern Web Applications 2nd Edition (2010).